Made for Each Other: Broad-coverage Semantic Structures Meet Preposition Supersenses

Jakob Prange   **Nathan Schneider**
Georgetown University
jakob@cs.georgetown.edu
nathan.schneider@georgetown.edu

Omri Abend
The Hebrew University of Jerusalem
oabend@cs.huji.ac.il

Abstract

Universal Conceptual Cognitive Annotation (UCCA; Abend and Rappoport, 2013) is a typologically-informed, broad-coverage semantic annotation scheme that describes coarse-grained predicate-argument structure but currently lacks semantic roles. We argue that lexicon-free annotation of the semantic roles marked by prepositions, as formulated by Schneider et al. (2018), is complementary and suitable for integration within UCCA. We show empirically for English that the schemes, though annotated independently, are compatible and can be combined in a single semantic graph. A comparison of several approaches to parsing the integrated representation lays the groundwork for future research on this task.

1 Introduction

A common thread in many approaches to meaning representation is the idea that abstract structures can describe semantic invariants that hold across paraphrasing or translation: for example, semantic dependency relations capturing predicate-argument structures or other types of semantic relations that can be annotated within sentences (e.g., Böhmová et al., 2003; Oepen et al., 2015; Banarescu et al., 2013). These annotation schemes can be distinguished by various design principles such as language-specificity; the level of granularity of meaning elements; the reliance on morphosyntactic criteria to define the units of semantic annotation; the extent to which human annotators specify semantics from scratch; and many others (Abend and Rappoport, 2017).

In this work, we seize an opportunity to unite two previously unrelated—yet complementary—meaning representations in NLP. On the one hand, Universal Conceptual Cognitive Annotation (UCCA; Abend and Rappoport, 2013) provides
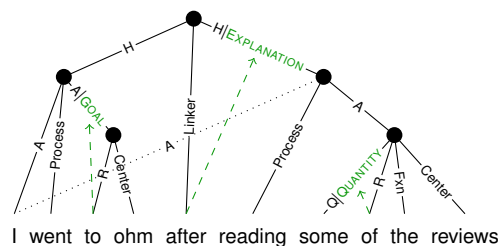

Figure 1: Semantic parse illustrating the integrated representation proposed here. Solid edges are the UCCA parse's *primary edges*, and the dotted edge is a *remote edge*. Dashed arrows show how SNACS labels (green small caps) have been mapped onto edges of the UCCA structure from the prepositions on which they were originally annotated. The following UCCA categories are abbreviated: A = Participant, R = Relator, H = Parallel scene, Q = Quantifier, Fxn = Function.

a skeletal structure of semantic units and relations, with typologically-based criteria for marking predicate-argument structures, based on "Basic Linguistic Theory", an established framework for typological description (Dixon, 2010/2012). On the other hand, a recent approach to annotation of English prepositions and possessives (SNACS; Schneider et al., 2018) provides an inventory of labels that characterize semantic relations. UCCA and SNACS follow similar design principles: they are both *language-neutral*, with general-purpose coarse-grained labels rather than lexically-specific senses or roles; and they are both designed for *direct* semantic annotation, without requiring a syntactic parse as a foundation. The philosophy is that these properties will facilitate annotation in many languages and domains that may lack detailed lexicons. But UCCA makes only the most rudimentary role distinctions, while SNACS annotations thus far have not made explicit which elements are being brought into a semantic relation (§2).

We propose a design that achieves the best of both worlds, as illustrated for an English example in figure 1. Taking advantage of an English corpus

that has been separately annotated for both UCCA and SNACS, as well as dependency syntax, we show that the SNACS role labels can be automatically integrated within UCCA structures over a range of syntactic constructions (§3). Then, we use this corpus to test pipelined, multitask, and joint approaches to parsing the integrated representation (§4). Our findings (§5) set the stage for future English parsers as well as multi- and cross-lingual extensions. §6 situates this work in the broader landscape of computational meaning representations.

Our main contributions are:

- a typologically-oriented broad-coverage linguistic **representation** that captures predicate-argument structure and semantic roles, without reference to any lexicon;
- a procedure to **integrate** UCCA and (token-level) SNACS annotations for particular sentences, mapping the SNACS labels to the appropriate edge in the UCCA structure, by which we create an integrated gold standard; and
- initial results for the **integrated parsing task**, comparing several alternatives that couple the learning/prediction of UCCA and SNACS in various ways. We find that optimizing for the two objectives jointly works best.

Data and code from these experiments are open-sourced to facilitate future work on this task.[1]

2 Background

To better understand the distinctions that we expect to be captured by such a framework, consider the following examples:

(1) a. Her picture was on the wall.
    b. Her speech was on security.

Despite parallel syntax and overlapping vocabulary, the sentences above vary in numerous aspects of meaning:

- The NPs *her picture* and *the wall* denote entities that stand in a certain locative relation to each other, as signaled by the preposition *on*.
- In contrast, the relation between *her speech* (which is an event, not an entity) and *security* is a different one, TOPIC, despite being signaled by

the same preposition.

This is made explicit in the German translations of these sentences:

(2) a. Ihr Bild   hing **an** der Wand .
       Her picture hung **at** the wall   .

   b. Ihre Rede war **über** Sicherheit .
      Her speech was **over** security   .

In addition, the possessive pronoun *her* (*ihr/ihre*) signals a prototypical POSSESSION relation in (1a)/(2a), but the core role of AGENT in (1b)/(2b).

As we can see, the natural lexical choices for expressing the LOCATION relation between the picture and the wall and the TOPIC relation in German have diverging literal translations to English. Thus, the empirical study of cross-linguistic commonalities and differences between form and meaning calls for a common metalanguage to describe the relations between mentioned events and entities, as marked by case and adpositions.

Our approach to such a representation consists of utilizing two existing semantic representations. UCCA (§2.1) captures the structure of predicate-argument and head-modifier relations at a high level, crucially distinguishing units that evoke a **scene** (event or state) from other units. SNACS (§2.2) disambiguates semantic roles as signaled by adpositions and possessives, but only directly annotates a function word, without formalizing the semantic relation that it mediates.[2] Both of these schemas have the guiding principle to be language-independent, eschewing a lexicon and defining a closed inventory of semantic categories.

2.1 UCCA

UCCA is a semantic annotation scheme rooted in typological and cognitive linguistic theory. It aims to represent the main semantic phenomena in the text, abstracting away from syntactic forms. UCCA's foundational layer, which is the only layer annotated over text so far,[3] reflects a coarse-grained level of semantics that has been shown to be preserved remarkably well across translations (Sulem et al., 2015). It has also been successfully used for improving text simplification (Sulem et al., 2018b),

---

[2]Note that mapping between syntactic and semantic relations varies by construction: in *She spoke on security*, the semantic head of the relation between *spoke* and *security* corresponds to the syntactic head (the verb), whereas in *Her speech was on security*, UD treats *security* as the syntactic head (§3).

[3]Prange et al. (2019) have proposed and piloted a coreference layer that sits above the foundational layer.

as well as to the evaluation of a number of text-to-text generation tasks (Birch et al., 2016; Sulem et al., 2018a; Choshen and Abend, 2018).

Formally, UCCA structures are directed acyclic graphs over **units** (nodes covering a subset of tokens). Atomic units are the leaves of the graph: individual tokens or unanalyzable MWEs. Nonterminal units represent larger semantic constituents, such as scenes and compositional participants/modifiers. The example in figure 1 has 5 nonterminal units. Each unit (save for the root) has a single incoming **primary** edge, and may also have incoming reentrant **remote** edges to express shared argumenthood. The primary edges of a UCCA structure thus form a tree, which along with the remote edges, forms a DAG.[4]

Edges are labeled with one or more **categories** indicating a kind of semantic relationship. The small set of categories includes State and Process for static or dynamic scene predicates, respectively; Participant, Time, and Adverbial for dependents of scenes; Center for the head of a non-scene unit (usually an entity); and Elaborator and Quantity for modifiers of entities. Scenes can be semantic dependents (Participant of another scene, Elaborator of a non-scene). Multiple scenes at the same level are called Parallel Scenes, and connectives between them are Linkers.

UCCA makes a distinction between different functions of prepositions, the most common cases of which are: (1) phrasal verbs (e.g., "give **up**"), annotated as internally unanalyzable; (2) linkers between scenes; e.g., in figure 1, "after" links the going to ohm scene, and the reading scene); (3) main relations in scenes (e.g., "The apple tree is **in** the garden"); and (4) case markers within a scene or a participant, or *Relators* in UCCA terms (e.g., in figure 1, "to" and "of" are such markers).

However, apart from distinguishing temporal modifiers, the UCCA scheme does not provide any **semantic role** information: thus the analyses of "the dark wizard defeated by Gandalf" and "the dark wizard's defeat of Gandalf" are nearly isomorphic—obliterating the distinction between agents and patients in the semantics—though the grammatical encoding of the noun phrases in question (subject, by-PP, possessive, of-PP) leaves no ambiguity about the intended roles to a human reader.

2.2 SNACS

SNACS is an inventory of 50 roles/relations used to disambiguate adpositions and possessives in multiple languages, including English (Schneider et al., 2018, 2019), Mandarin Chinese (Zhu et al., 2019), and to a lesser extent, Korean, Hindi, and Hebrew (Hwang et al., 2017). Many of the SNACS labels, such as AGENT, THEME, and TOPIC, are derived from VerbNet's (Kipper et al., 2008) core roles of predicates. (Others, such as QUANTITY and WHOLE, are for entity modification.) But unlike VerbNet, FrameNet (Fillmore and Baker, 2009), and PropBank (Palmer et al., 2005), SNACS does not require a language-specific predicate lexicon (hence Schneider et al. (2018) use the term "super-senses", which we adopt in the remainder of this paper)—and is therefore compatible with UCCA's design principle of crosslinguistic applicability.[5]

Currently, SNACS labels are applied directly to lexical items, without marking up underlying structure on either the subword (morphological) or the sentence-structure level.

3 Automatically Integrating Semantic Roles with UCCA

With the benefit of a jointly annotated corpus, we examined the data and determined that the proper placement of adpositional semantic role labels is fairly deterministic given certain syntactic patterns and their structural counterparts in UCCA. Here we present a rule-based method for automatically integrating token-based semantic role annotations from SNACS into an UCCA graph as edge refinements. We use these rules to construct a gold standard for analysis and parsing of the integrated representation. The rules we use, though empirically grounded in the English Web Treebank corpus (Bies et al., 2012), make no specific assumptions about language or lexicon, as they solely depend on UCCA, SNACS, and Universal Dependencies (UD; Nivre et al., 2016) annotation, all of which are frameworks designed to be cross-linguistically applicable. Thus, we expect the rules could be adapted to other languages with only minor changes if the underlying annotations are applied consistently, though this will require testing in future work.

---

[4]There is also the capability to annotate implicit units, but these are ignored in the standard evaluation and we do not address them here.

[5]SNACS also annotates the **function** of a preposition token—its lexical semantics which may be distinct from its semantic role (Hwang et al., 2017). Only scene roles are taken into account in the present analysis.

	scene	non-scene
verb	I _went_ [**to** ohm]	_Quit_ [**with** the overstatements] !
noun	Wonderful _service_ [**for** large group] [10 minutes] **of** _paperwork_	Cheapest _drinks_ [**in** Keene] [No amount] **of** _sugar_ and milk can mask it .

Table 1: Syntactic and semantic dimensions of canonical adpositional phrase constructions. The adposition is **bolded**, the semantic head is _italicized_, and the semantic dependent is [bracketed]. Rows indicate whether the semantic head is nominal or verbal, while columns differentiate between scene-evoking and non-scene-evoking heads. Scene-evokers are underlined.

3.1 Data

We use the STREUSLE 4.0 corpus (Schneider and Smith, 2015; Schneider et al., 2018), which covers the reviews section from the English_EWT treebank of UD 2.3, and lexical semantic annotations for the same text.[6] The same corpus has been annotated with UCCA by Hershcovich et al. (2019a). We use the standard train/dev/test split for this dataset (table 2). §3.3 shows the distribution of linguistic phenomena at issue here.

3.2 Procedure

Given an UCCA graph with annotated terminals, the integration routine projects a SNACS annotation of a token onto the appropriate edge representing a semantic relation. This is illustrated by dashed arrows in figure 1. The procedure starts with a single terminal node, traversing the graph upwards until it finds an edge that satisfies the criteria given by the rules. The rules concern canonical prepositional phrase modifiers, plus a variety of syntactically or otherwise anomalous constructions, such as copulas and adverbs.

3.2.1 Canonical PPs

The adpositional constructions annotated in STREUSLE can be adnominal or adverbial modifiers and arguments, where both the nouns and verbs that are being elaborated on can evoke either scenes or non-scene units in UCCA (table 1). First, we take a look at expressions marked with Relators in UCCA, which generally correspond to prototypical syntactic PPs.

Modifiers of scenes. In general, if the adposition marks a modifier of a scene—i.e., the adposition

---

	train	dev	test	total
sentences	2,723	554	535	3,812
tokens	44,804	5,394	5,381	55,579
SNACS-annotated	4,522	453	480	5,455
successful integ.	4,435	447	473	5,355
matches synt. obj	3,924	403	438	4,765

Table 2: Quantitative analysis of adpositional constructions in the corpus.

is the first or last terminal in the modifier unit's yield—the supersense should refine the role of that dependent. The adposition's parent unit is refined by the supersense ("to ohm" in figure 1 and table 1; "for large group" in table 1).

Modifiers of non-scenes. Where the adposition relates a modifying (elaborating or quantifying) unit to a non-scene unit, the supersense refines the modifying unit. If the adposition is the first or last terminal in a non-Center unit, that unit gets refined ("in Keene" in table 1). This includes the case when the adposition marks the predicate evoking the scene of which the syntactic governor is a modifier ("with the overstatements": "Quit" is treated as an aspectual modifier of the "overstatements" event).

For partitive constructions like "the top **of** the mountain", both the syntactic governor and object of the adposition are marked as UCCA Centers, indicating that they are on a _semantically equal_ level—neither one is clearly only modifying or being modified by the other. In this case, the supersense refines the syntactic object of the adposition.

Quantities. Another special case is where the adposition is labeled as QUANTITY, in which case the unit for its syntactic governor[7] ("10 minutes" in table 1) receives the refinement: e.g., "[some] **of** the reviews" in figure 1; in table 1, "[10 minutes] **of** paperwork" and "[no amount] **of** sugar" (the bracketed expression is the QUANTITY).

3.2.2 Non-canonical phenomena

For other, less prototypical constructions involving SNACS-annotated expressions, such as copulas, linked parallel scenes, possessive pronouns, and idiomatic PPs, we have additional rules, summarized in supplementary material (appendix A).

---

	train	dev	test	total
total primary edges	54,204	6,628	6,623	67,455
total remote edges	2,881	349	387	3,617
refined	4,473	449	479	5,401
≥ 1 edge refined	38	2	6	46
remote edges	33	2	5	40
canonical	2,468	242	270	2,980
scene mod	2,124	219	254	2,597
non-scene mod	344	23	16	383
non-canonical	2,005	207	209	2,421
predication	167	19	23	209
linkage	461	54	41	556
intransitive adp.	261	19	23	303
scn-mod nscn-mod	189 72	14 5	20 3	223 80
approximator	14	0	1	15
possessive pron.	897	81	95	1,073
scn-mod nscn-mod	774 123	72 9	87 8	933 140
infinitival	66	18	11	95
scn-mod nscn-mod	15 51	0 18	1 10	16 79
PP idiom	139	16	15	170

Table 3: Refined UCCA edges by construction type, according to our heuristic. (The non-canonical subcategories are mutually exclusive.)

3.3 Quantitative Analysis

We run the integration routine on our dataset and report statistics in tables 2 and 3. The heuristic rules have a coverage of 98% – 99% (row 'successful integ.' divided by row 'SNACS-annotated' in table 2). 88.5% (train) – 92.6% (test) of refined units contain the syntactic complement[8] (table 2), indicating that while syntax may often give a good approximation to the semantic relations marked by adpositions, a direct mapping from syntactic into semantic structure is not always trivial.

In table 3, we see that among the canonically adpositional SNACS targets, the vast majority mark scene modifiers (including participants). The various non-canonical targets modify both scenes and non-scenes, except for linkages and predications, which naturally only operate at the scene-level, and approximators, which only elaborate on non-scenes. Similar to canonical adpositional constructions, possessive pronouns and intransitive adpositions have a tendency to modify scenes rather than non-scenes. Those infinitivals that are not inter-scene Linkers (which are counted under *linkage*) are mostly non-scene modifiers.

Remote edges are only rarely affected by the SNACS integration, so we exclude them from evaluation in §5.

---

[8]We use the term 'syntactic complement' to include head tokens of prepositional objects and subordinate clauses, as determined by the script mentioned above. If no prepositional object is given in the STREUSLE corpus, we consider the adposition itself and only count towards this metric if the refined edge is its incoming preterminal edge.

3.4 Difficult Cases & Limitations

Our heuristics are based solely on universal semantic and syntactic annotations, with no assumptions about the grammar or lexicon of any specific language. However, there are some limitations to the rules that deserve discussion. Most importantly, as many rules are highly sensitive to UCCA structure and categories, errors or inconsistencies in human and automatic UCCA annotation are likely to throw the system off. This can be mitigated with strict constraints and careful reviewing during manual annotation, but cannot be fully avoided when applying the rules to automatically generated UCCA.

Multiword expressions (MWEs) are another source of difficulty. Both UCCA and STREUSLE mark idiomatic MWEs, but follow slightly different guidelines. The heuristic rules actually recover from most MWE misalignments; however, constructions that are MWEs in UCCA and contain multiple SNACS targets, such as *as-as* comparatives, are not fully resolved by our heuristic, as we cannot assign individual edge refinements for the adpositions' competing supersenses, given that we start traversing the UCCA graph from the shared preterminal node.

4 Models

We hypothesize that our combined (lexical and structural) semantic representation is not only linguistically plausible, but also useful in practice, as the annotations on both levels should be informative predictors for each other. That is, we expect that knowing what semantic role is signaled by an adposition informs the underlying semantic structure of the sentence, and vice versa.

In order to test this hypothesis, we use our annotated corpus to parse into the integrated representation. We consider several different ways of orchestrating the prediction of the foundational UCCA structure and the prediction of SNACS roles: modeling SNACS and UCCA in (i) a pipeline, (ii) a multitask setup with shared parameters, (iii) a single joint model.

4.1 Baseline: TUPA

We choose the neural transition-based graph parser TUPA (Hershcovich et al., 2017, 2018) as a strong baseline for UCCA parsing. It was the official baseline in the recent SemEval shared task on UCCA parsing (Hershcovich et al., 2019b).

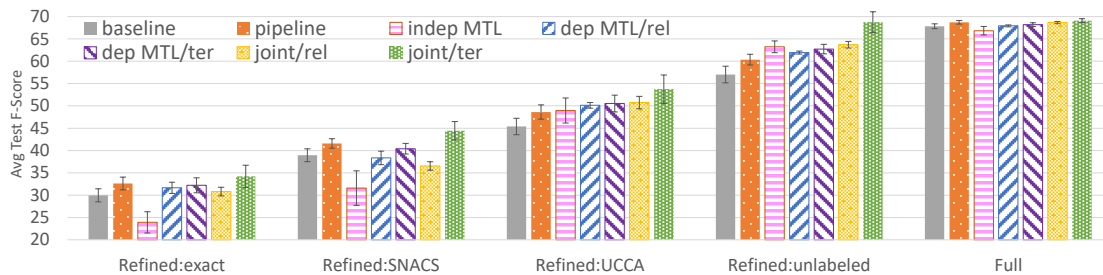TUPA's transition system is defined to address

Figure 2: Average F1-score on the test set over 5 random restarts with error bars indicating standard deviation. *ter* stands for terminal-level and *rel* for relation-level SNACS refinement (prediction or features).

the different formal structural phenomena exhibited by UCCA structures, notably reentrancies and discontiguous units. There are transitions for creating nonterminal nodes, and for attaching terminal and nonterminal nodes to a primary or remote (reentrant) parent with an UCCA category label on the edge. The transition system is general enough to be able to tackle parsing into a variety of formalisms, including SDP (Oepen et al., 2015) and a simplified form of AMR (Banarescu et al., 2013); Hershcovich et al. (2018) take advantage of this flexibility in their multitask learning framework.

TUPA's learning architecture largely follows that of Kiperwasser and Goldberg (2016). It encodes the parser's configuration (buffer, stack and intermediate graph structure) using BiLSTMs, and predicts the next transition using an MLP, stacked on top of them. Token-based features, including POS tags, dependency parses, as well as NER and orthographic features, are embedded in the BiLSTM. Another set of features, taking into account the partially constructed graph and previously predicted transition types, is fed into the MLP directly.

4.2 Pipeline

We extend TUPA by providing the SNACS label as a feature on the adposition token.[9] This is added in preprocessing in the same way as the syntactic features listed above (including the BiLSTM encoding). At testing time, we obtain SNACS labels for automatically identified targets from the SVM model of Schneider et al. (2018).

4.3 Multitask

Hershcovich et al. (2018) showed that UCCA parsing performance can be improved with multitask

learning (MTL; Caruana, 1997) on several semantic and syntactic parsing tasks. We examine whether alternately optimizing two objectives, one for UCCA and one for SNACS, leads to mutually favorable biases via shared parameters. There are multiple ways the two tasks can be orchestrated:

Independent MTL. This is the multitask learning (MTL) setup from Hershcovich et al. (2018), where separate transition classifiers are trained on different tasks simultaneously, sharing and mutually updating the BiLSTM encoding.[10] We consider as auxiliary tasks (a) SNACS scene role classification and (b) the decision of which UCCA unit is refined by a SNACS-annotated token. We encode these tasks as parsing tasks analogous to UCCA parsing as follows: for each training item in (a), we create a graph consisting of a root and up to 4 children: the syntactic governor (if available), the preposition token, the syntactic object (if available)—all of which have dummy edge labels—as well as a dummy terminal carrying the SNACS supersense. For each training item in (b), we consider the full UCCA structure, but the edge labels are simply boolean values indicating whether an edge is refined or not.

We also train a separate model with SNACS classification as the primary task and UCCA parsing and SNACS integration as auxiliary tasks, whose predictions are integrated in postprocessing for the combined evaluation (table 4), and which is evaluated independently in table 5.

Dependent MTL. Here we train the SNACS task in direct interaction with the UCCA parsing task.

---

[9]We report here only results for the setting in which a supersense is added as a feature of the *preposition* token. We also experimented with using it as a feature of the syntactic object token—which often, but not always, heads the semantic object (cf. table 2)—but got similar or worse results.

[10]Note that our setup differs from that of Hershcovich et al. (2018) in two key points: In contrast to the auxiliary tasks used in the aforementioned work, SNACS prediction as formulated by Schneider et al. (2018) is not a structured, but a (per-token) classification task (however, as described above, we transform it into an artificially structured task to make it conform with the input format expected by TUPA). Furthermore, we are interested in both UCCA and SNACS performance, expecting both tasks to benefit from each other's complementary semantic content.

system	ref	Refined: exact			Refined: SNACS			Refined: UCCA			Refined: unlabeled			Full		
setup	ref	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
BL	–	30.2	29.7	30.0	39.3	38.7	38.9	45.7	45.1	45.4	57.4	56.6	57.0	68.2	67.6	67.8
(oracle SNACS)		45.4	45.1	45.2	62.7	62.3	62.5	48.9	48.6	48.7	62.7	62.3	62.5	69.5	68.9	69.2
pipeline	ter	32.9	32.4	32.6	42.0	41.3	41.6	49.1	48.2	48.6	60.9	59.9	60.4	68.8	68.6	68.7
(oracle SNACS)		53.5	53.2	53.3	70.4	70.0	70.2	57.0	56.7	56.9	70.4	70.0	70.2	71.0	70.7	70.8
indep MTL	ter	26.0	22.2	23.9	34.3	29.3	31.6	53.1	45.4	49.0	68.7	58.6	63.2	67.1	66.6	66.8
dep MTL	ter	34.4	30.3	32.2	43.1	38.0	40.4	53.9	47.5	50.5	66.9	59.1	62.7	68.4	68.1	68.2
	rel	32.7	30.6	31.6	39.6	37.2	38.3	51.8	48.6	50.1	64.0	60.1	61.9	68.3	67.6	67.9
joint	ter	**34.5**	**34.3**	**34.2**	**44.6**	**44.3**	**44.4**	53.9	**53.5**	**53.7**	69.0	**68.5**	**68.7**	**69.5**	**68.7**	**69.1**
	rel	34.0	28.1	30.8	40.4	33.4	36.5	**56.1**	46.4	50.7	**70.3**	58.2	63.7	69.1	68.3	68.7

Table 4: Experimental results, averaged over 5 random restarts. The baseline system (BL) for UCCA is TUPA version 1.3.9 without any modifications, retrained on our data. For the sake of generalizability and consistency with our own preprocessing, we use system-predicted SNACS categories from the auto-id/auto-syntax setting from (Schneider et al., 2018) in the BL and pipeline setups. Results where the system has access to gold SNACS annotations on adposition tokens are shown in small font.

We enhance TUPA with a separate MLP that, given an edge, classifies its supersense refinement (a null category can be chosen to indicate an unrefined edge). This network is run after each edge-creating transition. Its input features are the same as for the transition classifier, including the BiLSTM encoding. Since the two classifiers alternate in making forward passes and updating the shared BiLSTM, they indirectly contribute to each other's input. Here we have an option of where in the UCCA structure to initially predict the supersense label. In the **terminal-level (ter)** setting, we predict supersense refinements only on preterminal edges, and then apply the integration rules (§3) as postprocessing. In the **relation-level (rel)** setting, we parse directly into the integrated representation. To do this, we preprocess the training data with our integration routine. However, during parsing, there is no explicit restriction that supersense-refined edges must have an adposition token in their yield—thus the model could, in theory, learn to predict adequate role supersenses even when it is not signaled by a lexical marker (though it will get penalized for that in our current evaluation).

4.4 Joint

Finally, we train a single classifier on the integrated data, concatenating UCCA and SNACS categories, to predict parsing transitions using the new compound categories. We revisit the terminal-level and the relation-level settings introduced in §4.3.

5 Experiments

5.1 Experimental Setup

Preprocessing. We follow Hershcovich et al. (2018) in obtaining automatic POS and NE tags, as well as syntactic dependency relations using

SpaCy 2.0, and pretrained word vectors from fast-Text.[11] For all setups that use or predict SNACS supersenses, we include the gold standard scene role categories for pre-identified targets from the STREUSLE 4.0 corpus in our training and development data. In the test data we identify adposition targets using the heuristics introduced in Schneider et al. (2018). For the joint prediction setup (§4.3), we also include the head terminal of the syntactic governor and object for each adposition as features, using the same heuristics as in §3.

Architecture and hyperparameters. For classifying the next transition, TUPA uses a multi-layer perceptron (MLP) with 2 hidden layers and a softmax output layer, on top of a 2-layer BiLSTM (Kiperwasser and Goldberg, 2016). Building on previous work, we train for 100 epochs, using the stochastic gradient descent optimizer for the first 50, and AMS-grad (Reddi et al., 2018) for the remaining 50 epochs,[12] and apply early stopping post-hoc by keeping the model with the highest performance on the dev set as the final model.

Evaluation. For our main evaluation in §5.2, we compare our systems along five new metrics: a **full structure** score which evaluates precision and recall of all units and requires both the UCCA and SNACS categories to be correct, where applicable; and **refined UCCA**, **SNACS**, **exact**, and **unlabeled** scores which only consider SNACS-refined units. Here, the integrated representation obtained via the rule-based integration (§3.2) serves as the ground truth. We also report the standard labeled and unlabeled UCCA scores.[13] In addition, for sys-

---

[11] https://spacy.io/; https://fasttext.cc/
[12] Except for the independent MTL setting, where we stop training after the first 50 epochs.
[13] All of the above metrics are F-scores over the edges, as in Hershcovich et al. (2017).

system setup	ref	UCCA labeled			UCCA unlabeled			SNACS		
		P	R	F	P	R	F	P	R	F
BL	–	72.5 ±0.6	71.9 ±.4	72.2 ±0.4	88.5 ±0.3	87.6 ±.5	88.0 ±.3	58.5	58.3	58.4
pipeline (oracle SNACS)	ter	72.4 ±0.3	**72.0** ±.6	72.2 ±0.4	88.3 ±0.2	**87.8** ±.6	88.1 ±.4	–	–	–
		73.0 ±0.4	72.6 ±.7	72.8 ±0.5	88.8 ±0.2	88.3 ±.7	88.5 ±.4	–	–	–
indep MTL	ter	71.0 ±1.2	70.2 ±.9	70.6 ±1.0	87.9 ±1.0	86.9 ±.8	87.3 ±.7	48.2 ±5.6	41.4 ±4.8	44.6 ±5.2
dep MTL	ter	71.8 ±0.4	71.3 ±.2	71.5 ±0.3	88.1 ±0.4	87.6 ±.2	87.8 ±.2	60.1 ±1.8	53.3 ±2.0	56.5 ±1.9
	rel	71.8 ±0.3	71.0 ±.1	71.4 ±0.1	88.0 ±0.3	87.0 ±.4	87.5 ±.3	–	–	–
joint	ter	**72.8** ±0.3	71.9 ±.4	**72.3** ±0.3	**88.7** ±0.3	87.6 ±.4	**88.2** ±.3	**60.5** ±3.0	60.3 ±3.4	**60.4** ±3.2
	rel	72.5 ±0.4	71.5 ±.2	72.0 ±0.3	88.5 ±0.2	87.2 ±.3	87.8 ±.2	–	–	–

Table 5: Results on the respective tasks of UCCA parsing and token-level SNACS prediction, averaged over 5 random restarts, with standard deviation reported next to each average. The baseline system (BL) for UCCA is TUPA version 1.3.9 without any modifications, retrained on our data. The SNACS baseline system is the SVM classifier of Schneider et al. (2018).

tems which predict a terminal-level SNACS label (before it is mapped to a higher relation in post-processing), we compare SNACS disambiguation performance against (Schneider et al., 2018) in §5.3.

5.2 Integrated parsing results

Our MTL and joint systems outperform the baseline and a feature pipeline on refined UCCA units (figure 2 and table 4). The main benefit from considering UCCA and SNACS together in training is that the parser is better at recovering the (unlabeled) structure of units that should receive a SNACS relation in the integrated representation. This is illustrated in figure 3. This trend is confirmed in the precision and recall of UCCA units that have a gold SNACS token in their yield (unlabeled F-score: *BL* = 93.1, *dep-MTL/ter* = 95.2, *indep-MTL* = 96.0, see table 7 in the supplementary material). To the extent that these units are syntactic constituents (see table 2), this suggests that multitask learning with syntactic auxiliary tasks (Swayamdipta et al., 2018; Hershcovich et al., 2018) might be particularly beneficial for SNACS-augmented UCCA parsing. The feature pipeline is competitive, but noisy features from a previous classification step limit its performance on refined units. The upper bounds given by the oracle setting indicate that SNACS features are generally beneficial. *Indep-MTL* and systems that parse directly into the relation-refined representation struggle with predicting the correct SNACS refinements—and thus also exact UCCA+SNACS combinations—while the *joint/ter* model is consistently the most accurate.

However, there is little effect on overall labeled and unlabeled UCCA scores (table 5). Predicting SNACS simultaneously or interactively with UCCA (*joint/rel* and *dep-MTL*) apparently makes the parsing task harder. Note that particularly in
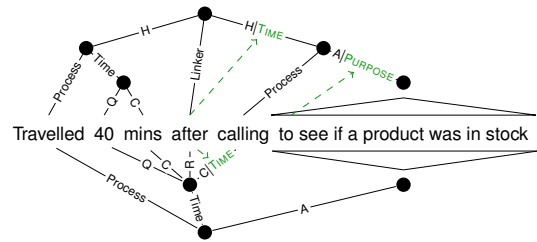

Figure 3: Simplified sample output. The *joint/ter* system (top) generates the intended scene structure and PURPOSE modifier attachment. The *BL* system (bottom) does not, and misses the PURPOSE role altogether.

setup	ref	# params	
BL	–	78.9M	
pipeline	ter	78.9M	
indep MTL	ter	**UCCA**	**SNACS**
		82.8M	81.8M
dep MTL	ter	79.4M	
	rel	79.4M	
joint	ter	79.2M	
	rel	79.2M	

Table 6: Number of parameters of each model.

the *dep-MTL* setting, erroneous decisions in one task could negatively affect the other.

5.3 Token-based SNACS prediction results

Since some of our systems predict SNACS labels at the terminal level, they are directly comparable to previous work on SNACS classification. We compare against the auto-id/auto-syntax baseline from Schneider et al. (2018) in table 5.[14] Both the *dep MTL* and *joint* systems outperform the baseline in precision; and the *joint* system also in recall, leading to the overall best performance. The *indep-MTL* system does not reach baseline performance.

5.4 Model capacity

We examine whether the differences in performance can really be attributed to the linguistic in-

---

[14] Due to the diverging guidelines on multiword units in UCCA (§3.4), we ignore MWE boundaries.

formation in our data or merely to more powerful models by inspecting the number of each model's parameters (table 6). While we observe some variance in model capacity, we consider these to be minor differences. An exception is the *independent MTL* setup, which consists of two independent models, each dedicated to a specific task. However, this does not seem to give it an advantage in terms of final performance. The baseline has the fewest parameters, and the overall best condition, *joint/ter*, is neither the smallest nor the largest model, suggesting that the particular linguistic signals and the method of using them have a genuine effect on performance.

6 Related Work

The benefits of integrating lexical analysis and sentence semantic or syntactic structure have been pursued by a vast body of work over the years. Compositional approaches to the syntax-semantics interface, such as CCG (Steedman, 2000) and HPSG (Pollard and Sag, 1994), usually integrate the lexicon at the leaves of the syntactic parse, but propagate grammatically-relevant features up the tree. A different approach is taken by OntoNotes (Hovy et al., 2006), which consists of a number of separate, albeit linked tiers, including syntactic and argument structure, but also the lexical tiers of word senses and coreference.

Role semantics frequently features in structured semantic schemes. Some approaches, such as PropBank and AMR (Palmer et al., 2005; Banarescu et al., 2013), follow a lexical approach. The Prague Dependency Treebank tectogrammatical layer (Böhmová et al., 2003) uses a few lexicon-free roles, but their semantics is determined by virtue of their linkage to a lexicalized valency lexicon. Universal Decompositional Semantics (White et al., 2016) instead defines roles as a bundle of lexicon-free features, elicited by crowdsourcing.

The specific inventory for preposition/possessive relations that we use is SNACS, but there is a wider history of disambiguation of these items, especially in English: disambiguation systems have been described for possessives (Moldovan et al., 2004; Badulescu and Moldovan, 2009; Tratz and Hovy, 2013), prepositions with lexicalized sense definitions (e.g., Litkowski and Hargraves, 2007; Tratz and Hovy, 2011), and prepositions with coarse-grained classes (O'Hara and Wiebe, 2003, 2009; Srikumar and Roth, 2013; Gonen and Goldberg,

2016). Such disambiguation has also been investigated in tandem with semantic role labeling and parsing (Dahlmeier et al., 2009; Srikumar and Roth, 2011; Gong et al., 2018). Preliminary work suggests that SNACS may be applicable to subjects and objects, not just PPs, and thus in the future this framework could be extended to all UCCA participants (Shalev et al., 2019).

State-of-the-art results on UCCA parsing and SNACS disambiguation are described in contemporaneous work by Jiang et al. (2019); Liu et al. (2019), who achieve substantial gains using the ELMo and BERT contextualized word embeddings (Peters et al., 2018; Devlin et al., 2019). This is an orthogonal direction to the one we pursue here, and combining the two is left to future work.

7 Conclusion

We have introduced a new representation combining UCCA semantic structures and SNACS adpositional semantic roles; automatically merged existing annotations to create a gold standard; and experimented with several alternatives for parsing the integrated representation. Our results show that models profit from having access to both structural and lexical semantic information, confirming our hypothesis that UCCA and SNACS are complementary and compatible.

Based on preliminary results from a German corpus, we conjecture that this approach is applicable to other languages with no or only minimal changes—a direction we will explore further in future work. In addition, we plan to investigate the utility of the enhanced representation for downstream tasks involving meaning-preserving linguistic variation.

References

Omri Abend and Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). In *Proc. of ACL*, pages 228–238, Sofia, Bulgaria.

Omri Abend and Ari Rappoport. 2017. The state of the art in semantic representation. In *Proc. of ACL*, pages 77–89, Vancouver, Canada.

Adriana Badulescu and Dan Moldovan. 2009. A Semantic Scattering model for the automatic interpretation of English genitives. *Natural Language Engineering*, 15(2):215–239.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proc. of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria.

Ann Bies, Justin Mott, Colin Warner, and Seth Kulick. 2012. English Web Treebank. Technical Report LDC2012T13, Linguistic Data Consortium, Philadelphia, PA.

Alexandra Birch, Omri Abend, Ondřej Bojar, and Barry Haddow. 2016. HUME: Human UCCA-based evaluation of machine translation. In *Proc. of EMNLP*, pages 1264–1274, Austin, Texas.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2003. The Prague Dependency Treebank: A three-level annotation scenario. In Anne Abeillé, editor, *Treebanks: Building and Using Parsed Corpora*, Text, Speech and Language Technology, pages 103–127. Springer Netherlands, Dordrecht.

Rich Caruana. 1997. Multitask Learning. *Machine Learning*, 28(1):41–75.

Leshem Choshen and Omri Abend. 2018. Reference-less measure of faithfulness for grammatical error correction. In *Proc. of NAACL-HLT*, pages 124–129, New Orleans, Louisiana.

Daniel Dahlmeier, Hwee Tou Ng, and Tanja Schultz. 2009. Joint learning of preposition senses and semantic roles of prepositional phrases. In *Proc. of EMNLP*, pages 450–458, Suntec, Singapore.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT*.

Robert M. W. Dixon. 2010/2012. *Basic Linguistic Theory*. Oxford University Press.

Charles J. Fillmore and Collin Baker. 2009. A frames approach to semantic analysis. In Bernd Heine and Heiko Narrog, editors, *The Oxford Handbook of Linguistic Analysis*, pages 791–816. Oxford University Press, Oxford, UK.

Hila Gonen and Yoav Goldberg. 2016. Semi supervised preposition-sense disambiguation using multilingual data. In *Proc. of COLING*, pages 2718–2729, Osaka, Japan.

Hongyu Gong, Suma Bhat, and Pramod Viswanath. 2018. Embedding syntax and semantics of prepositions via tensor decomposition. In *Proc. of NAACL-HLT*, pages 896–906, New Orleans, Louisiana.

Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2017. A transition-based directed acyclic graph parser for UCCA. In *Proc. of ACL*, pages 1127–1138, Vancouver, Canada.

Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2018. Multitask parsing across semantic representations. In *Proc. of ACL*, pages 373–385, Melbourne, Australia.

Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2019a. Content differences in syntactic and semantic representation. In *Proc. of NAACL-HLT*, pages 478–488, Minneapolis, Minnesota.

Daniel Hershcovich, Zohar Aizenbud, Leshem Choshen, Elior Sulem, Ari Rappoport, and Omri Abend. 2019b. SemEval-2019 Task 1: Cross-lingual Semantic Parsing with UCCA. In *Proc. of SemEval*, pages 1–10, Minneapolis, Minnesota, USA.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: the 90% solution. In *Proc. of HLT-NAACL*, pages 57–60, New York City, USA.

Jena D. Hwang, Archna Bhatia, Na-Rae Han, Tim O'Gorman, Vivek Srikumar, and Nathan Schneider. 2017. Double trouble: the problem of construal in semantic annotation of adpositions. In *Proc. of *SEM*, pages 178–188, Vancouver, Canada.

Wei Jiang, Zhenghua Li, Yu Zhang, and Min Zhang. 2019. HLT@SUDA at SemEval-2019 Task 1: UCCA graph parsing as constituent tree parsing. In *Proc. of SemEval*, pages 11–15, Minneapolis, Minnesota, USA.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional LSTM feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A large-scale classification of English verbs. *Language Resources and Evaluation*, 42(1):21–40.

Ken Litkowski and Orin Hargraves. 2007. SemEval-2007 Task 06: Word-Sense Disambiguation of Prepositions. In *Proc. of SemEval*, pages 24–29, Prague, Czech Republic.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proc. of NAACL-HLT*, pages 1073–1094, Minneapolis, Minnesota.

Dan Moldovan, Adriana Badulescu, Marta Tatu, Daniel Antohe, and Roxana Girju. 2004. Models for the semantic classification of noun phrases. In *HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, pages 60–67, Boston, Massachusetts, USA.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: a multilingual treebank collection. In *Proc. of LREC*, pages 1659–1666, Portorož, Slovenia.

Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinková, Dan Flickinger, Jan Hajič, and Zdeňka Urešová. 2015. SemEval 2015 Task 18: Broad-Coverage Semantic Dependency Parsing. In *Proc. of SemEval*, pages 915–926, Denver, Colorado.

Tom O'Hara and Janyce Wiebe. 2003. Preposition semantic classification via Treebank and FrameNet. In *Proc. of CoNLL*, pages 79–86, Edmonton, Canada.

Tom O'Hara and Janyce Wiebe. 2009. Exploiting semantic role resources for preposition disambiguation. *Computational Linguistics*, 35(2):151–184.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: an annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL-HLT*, pages 2227–2237, New Orleans, Louisiana.

Carl Pollard and Ivan A. Sag. 1994. *Head-driven Phrase Structure Grammar*. University of Chicago Press.

Jakob Prange, Nathan Schneider, and Omri Abend. 2019. Semantically constrained multilayer annotation: the case of coreference. In *Proc. of the First International Workshop on Designing Meaning Representations*, pages 164–176, Florence, Italy.

Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. 2018. On the convergence of Adam and beyond. In *Proc. of ICLR*, Vancouver, British Columbia, Canada.

Nathan Schneider, Jena D. Hwang, Archna Bhatia, Vivek Srikumar, Na-Rae Han, Tim O'Gorman, Sarah R. Moeller, Omri Abend, Adi Shalev, Austin Blodgett, and Jakob Prange. 2019. Adposition and Case Supersenses v2.3: Guidelines for English. *arXiv:1704.02134v4 [cs]*.

Nathan Schneider, Jena D. Hwang, Vivek Srikumar, Jakob Prange, Austin Blodgett, Sarah R. Moeller, Aviram Stern, Adi Bitan, and Omri Abend. 2018. Comprehensive supersense disambiguation of English prepositions and possessives. In *Proc. of ACL*, pages 185–196, Melbourne, Australia.

Nathan Schneider and Noah A. Smith. 2015. A corpus and model integrating multiword expressions and supersenses. In *Proc. of NAACL-HLT*, pages 1537–1547, Denver, Colorado.

Adi Shalev, Jena D. Hwang, Nathan Schneider, Vivek Srikumar, Omri Abend, and Ari Rappoport. 2019. Preparing SNACS for subjects and objects. In *Proc. of the First International Workshop on Designing Meaning Representations*, pages 141–147, Florence, Italy.

Vivek Srikumar and Dan Roth. 2011. A joint model for extended semantic role labeling. In *Proc. of EMNLP*, pages 129–139, Edinburgh, Scotland, UK.

Vivek Srikumar and Dan Roth. 2013. Modeling semantic relations expressed by prepositions. *Transactions of the Association for Computational Linguistics*, 1:231–242.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press, Cambridge, MA.

Elior Sulem, Omri Abend, and Ari Rappoport. 2015. Conceptual annotations preserve structure across translations: a French-English case study. In *Proc. of the 1st Workshop on Semantics-Driven Statistical Machine Translation (S2MT 2015)*, pages 11–22, Beijing, China.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018a. Semantic structural evaluation for text simplification. In *Proc. of NAACL-HLT*, pages 685–696, New Orleans, Louisiana.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018b. Simple and effective text simplification using semantic and neural methods. In *Proc. of ACL*, pages 162–173, Melbourne, Australia.

Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyer, Chris Dyer, and Noah A. Smith. 2018. Syntactic scaffolds for semantic structures. In *Proc. of EMNLP*, pages 3772–3782, Brussels, Belgium.

Stephen Tratz and Eduard Hovy. 2011. A fast, accurate, non-projective, semantically-enriched parser. In *Proc. of EMNLP*, pages 1257–1268, Edinburgh, Scotland, UK.

Stephen Tratz and Eduard Hovy. 2013. Automatic interpretation of the English possessive. In *Proc. of ACL*, pages 372–381, Sofia, Bulgaria.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal Decompositional Semantics on Universal Dependencies. In *Proc. of EMNLP*, pages 1713–1723, Austin, Texas, USA.

Yilun Zhu, Yang Liu, Siyao Peng, Austin Blodgett, Yushi Zhao, and Nathan Schneider. 2019. Adpositional supersenses for Mandarin Chinese. In *Proc. of SCiL*, volume 2, pages 334–337, New York, NY, USA.